PIZZA SALES ANALYSIS

DIMPYBANGORIYA@GMAIL.COM

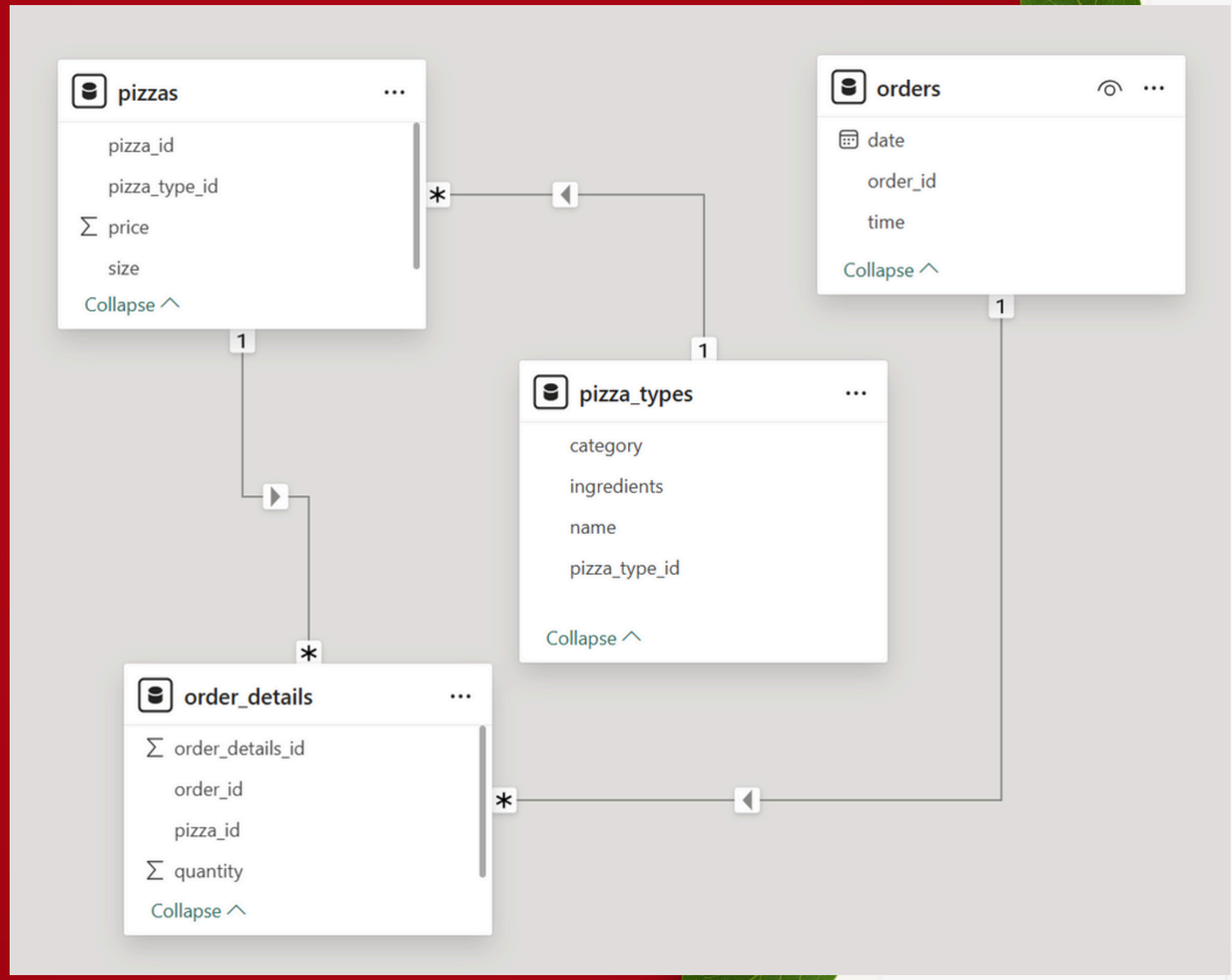
ABOUT THE PROJECT

This project focuses on analyzing pizza sales data using SQL to extract valuable business insights. Various SQL queries were used to fetch key information such as total sales, top-selling pizza types, customer order patterns, revenue trends, and peak sales hours. The analysis helps identify popular menu items, optimize inventory management, and improve sales strategies. By leveraging SQL for data retrieval and analysis, the project enables data-driven decision-making to enhance business performance and customer satisfaction.

SCHEMA

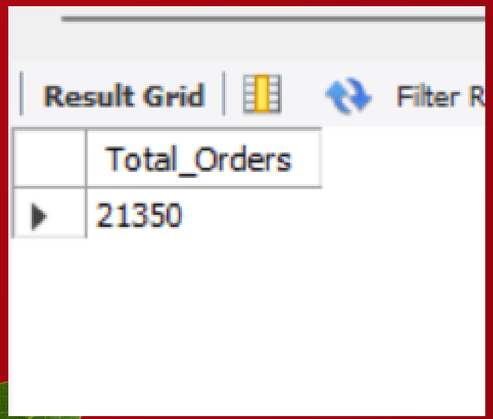
TABLES USED

PIZZA_TYPE
PIZZAS
ORDER_DETAILS
ORDERS



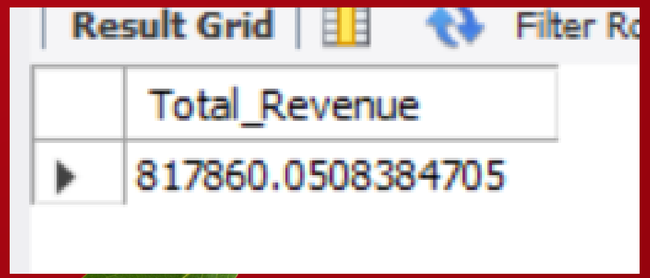
Q1. RETRIEVE TOTAL NO. OF ORDERS PLACED

```sql
-- 1 Retrieve the total number of orders placed.

select count(o1.order_id) as Total_Orders
from dominos.orders o1;
```



Q2. CALCULATE THE TOTAL REVENUE FROM PIZZA SALES

```sql
-- 2 Calculate the total revenue generated from pizza sales.

select sum(x.Revenue) as Total_Revenue from
(select (price*quantity) as Revenue
from dominos.order_details od1
left join dominos.pizzas p1
on od1.pizza_id = p1.pizza_id)x;
```



Q3. IDENTIFY THE HIGHEST PRICED PIZZA

```sql
-- 3 Identify the highest-priced pizza.

select t1.name,p1.price
from dominos.pizzas p1
inner join dominos.pizza_type t1
on p1.pizza_type_id = t1.pizza_type_id
where p1.price=(select max(price) from dominos.pizzas);
```

Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95

Q4. IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```sql
-- 4 Identify the most common pizza size ordered.


select p1.size, sum(od1.quantity) as Total_Pizza_By_Size
from dominos.order_details od1
left join dominos.pizzas p1
on p1.pizza_id = od1.pizza_id
group by p1.size
order by Total_Pizza_By_Size desc
Limit 1;
```

Result Grid | Filter Rows:

size	Total_Pizza_By_Size
L	18956

Q5. RETRIEVE TOP 5 MOST ORDERED PIZZA TYPES & THEIR QUANTITIES

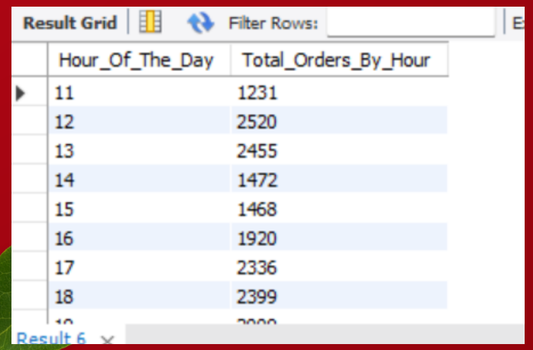
```
-- 5 List the top 5 most ordered pizza types
--  along with their quantities.

select p1.pizza_type_id,count(od1.quantity)
as total_orders
from dominos.order_details od1
inner join dominos.pizzas p1
on od1.pizza_id = p1.pizza_id
group by p1.pizza_type_id
order by total_orders desc
limit 5;
```

Result Grid | Filter Rows:

pizza_type_id	total_orders
classic_dlx	2416
bbq_ckn	2372
hawaiian	2370
pepperoni	2369
thai_ckn	2315

Q8. DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```sql
-- 6 Determine the distribution of orders
-- by hour of the day.

select hour(o1.order_time)
as Hour_Of_The_Day,
count(o1.order_id) as Total_Orders_By_Hour
from dominos.orders o1
group by hour(o1.order_time);
```



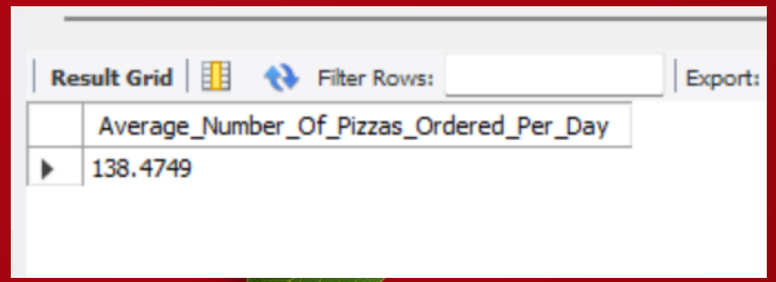
Result Grid | Filter Rows:

Hour_Of_The_Day	Total_Orders_By_Hour
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399

Result 6

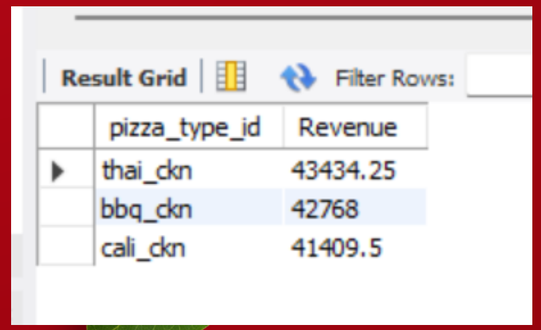
Q7. CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY

```sql
-- 7 calculate the average number of pizzas ordered per day.

select avg(x.Total_Quantity_Per_Day)
as Average_Number_Of_Pizzas_Ordered_Per_Day
from(
select o1.Order_date,sum(od1.quantity)
as Total_Quantity_Per_Day
from dominos.orders o1
inner join dominos.order_details od1
on od1.order_id = o1.order_id
group by o1.Order_date)x;
```



Q8. DETERMINE THE TOP 3 MOST ORDERED PIZZA

```sql
-- 8 Determine the top 3 most ordered
-- pizza types based on revenue.

select p1.pizza_type_id,
sum(p1.price*od1.quantity) as Revenue
from dominos.order_details od1
inner join dominos.pizzas p1
on od1.pizza_id = p1.pizza_id
group by p1.pizza_type_id
order by Revenue desc
Limit 3;
```

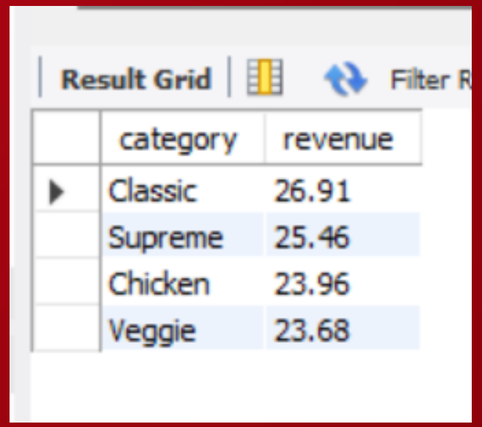


Result Grid | Filter Rows:

pizza_type_id	Revenue
thai_ckn	43434.25
bbq_ckn	42768
cali_ckn	41409.5

Q9. RETRIEVE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

```sql
-- 9 Calculate the percentage contribution
-- of each pizza type to total revenue.

select pizza_type.category,
Round(Sum(order_details.quantity * pizzas.price) /
(SELECT Round(sum(order_details.quantity * pizzas.price),02)
As total_sales
fromorder_details
JOIN
Pizzas
Where
order_details.pizza_id = pizzas.pizza_id)*100,2)
As revenue
From
pizza_type
JOIN
pizzas
on pizza_type.pizza_type_id = pizzas.pizza_type_id
join
order_details
on order_details.pizza_id = pizzas.pizza_id
group by pizza_type.category
order by revenue Desc;
```



Result Grid | Filter R

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

Q10 ANALYZE CUMULATIVE REVENUE GENERATED OVER TIME

```sql
-- 10 Analyze the cumulative revenue generated over time.


select y.Order_date, y.total_Revenue, sum(y.Total_Revenue)
 over(order by y.Order_date) as Cum_Revnue from
    (
    select x.Order_date, sum(x.revenue) as Total_Revenue
    from
            (
            select o1.Order_date, (p1.price*od1.quantity) as Revenue
            from dominos.pizzas p1
            left join dominos.order_details od1
            on p1.pizza_id = od1.pizza_id
            left join dominos.orders o1
            on o1.order_id = od1.order_id
            )x

        group by x.Order_date
    )y;
```

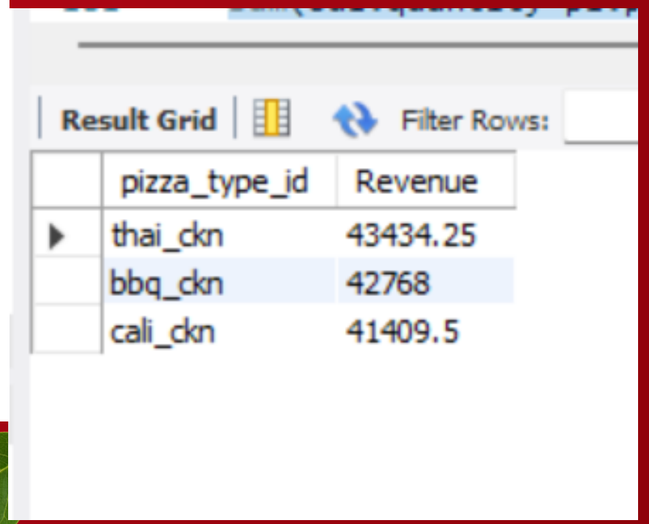
Result Grid | Filter Rows: | Export:

Order_date	total_Revenue	Cum_Revnue
NULL	NULL	NULL
2015-01-01	2713.8500022888184	2713.8500022888184
2015-01-02	2731.900001525879	5445.750003814697
2015-01-03	2662.4000034332275	8108.150007247925
2015-01-04	1755.4500007629395	9863.600008010864
2015-01-05	2065.9500007629395	11929.550008773804
2015-01-06	2428.950002670288	14358.500011444092
2015-01-07	2202.2000007629395	16560.70001220703

Result 10

Q11. DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY

```sql
-- 11 Determine the top 3 most ordered pizza types
-- based on revenue for each pizza category.


select pt1.pizza_type_id,
sum(od1.quantity*p1.price) as Revenue
from dominos.order_details od1
left join dominos.pizzas p1
on od1.pizza_id = p1.pizza_id
left join dominos.pizza_type pt1
on pt1.pizza_type_id = p1.pizza_type_id
group by pt1.pizza_type_id
order by Revenue desc
Limit 3;
```



Result Grid | Filter Rows:

pizza_type_id	Revenue
thai_ckn	43434.25
bbq_ckn	42768
cali_ckn	41409.5

CONCLUSION

This project successfully analyzed pizza sales data using SQL, providing valuable insights into sales performance, customer preferences, and business trends. By executing various SQL queries, key information such as total revenue, top-selling pizzas, peak sales hours, and customer ordering behavior was extracted. The findings help optimize inventory management, refine pricing strategies, and enhance marketing efforts. Overall, the project demonstrates the power of SQL in extracting actionable insights, enabling data-driven decision-making to improve sales and customer satisfaction in the food industry.

THANK YOU!

DIMPYBANGORIYA@GMAIL.COM